**Work Order System**

Technical Manual

CLASS: CPSC488

3/5/2023

Trevor Bell trevorbell030@gmail.com

Ellie Wurst

John T Friend Jr.

Hunter Minteer

1. Overview

This document will describe, in detail, what is needed and how to run this work order system project

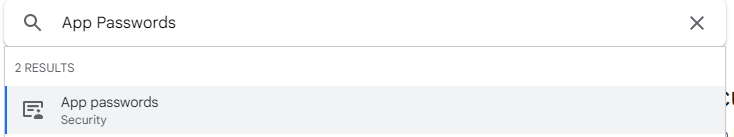
1. System Requirements
2. Assumptions

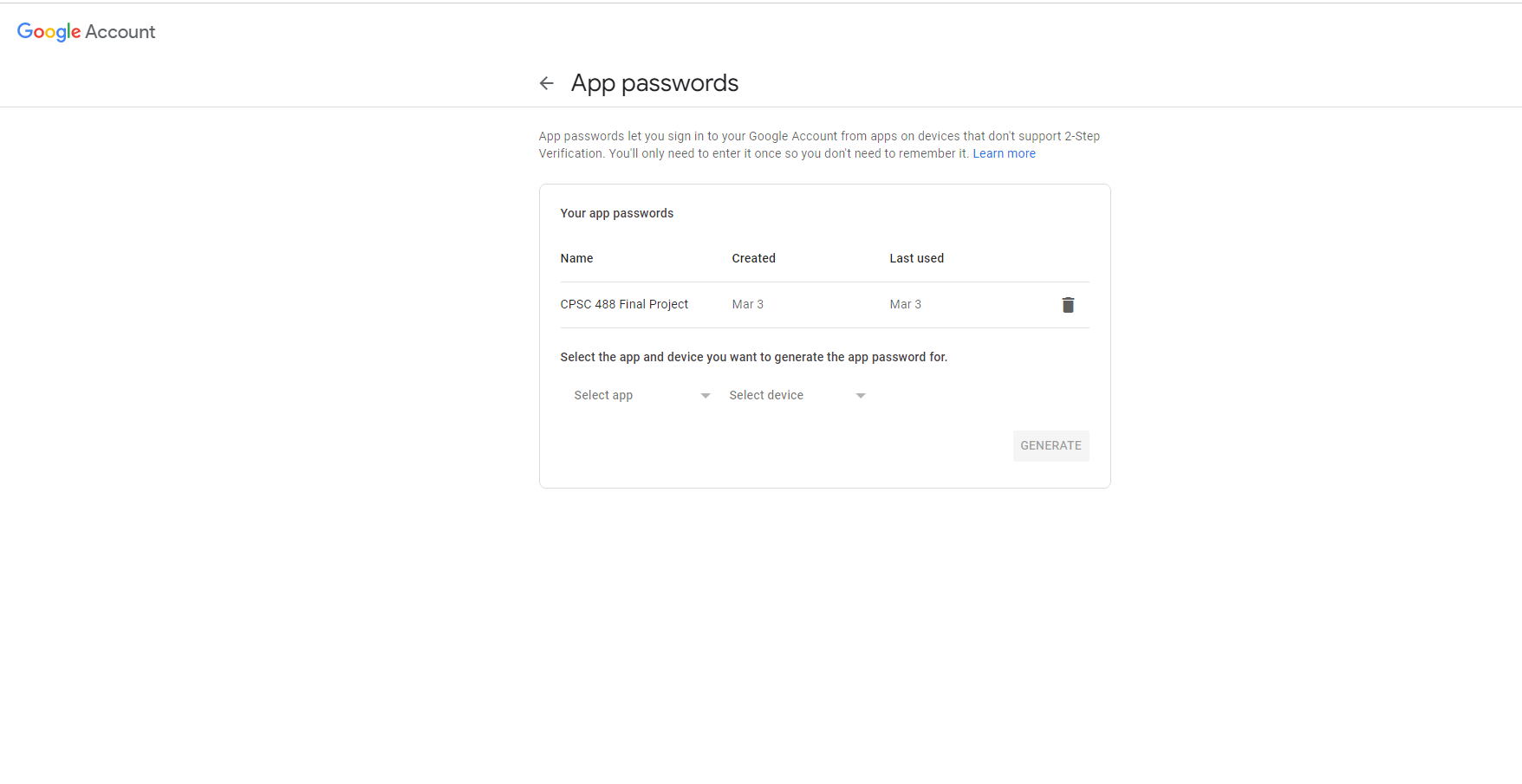
* This project assumes users will run on the latest version of Java using Eclipse IDE
* It also assumes running on a Windows 10 environment
* Access to the internet and Google Chrome internet browser

1. Setting up SMTP

For proper use of this project, emails are sent out to users when they register, send an incident report, and then they will also get notified when their incident is completed. And with this in mind a Gmail account will need to be made so that the emails have a source.

So, within the project there is a part of the application.properties file which will need the email username and password set up, but the password used normally both should not be used and will not work with the project. Here are the steps to get a proper application specific password for use with this system. Going to your account within the browser and typing “App Passwords and hitting the setting labeled as such with get you to the proper location





Hitting the “Select app” drop down and then hitting “other” will give you a text box in which you can create a name for the password, create the name, and then selecting generate will give you the key needed. The password will then show up in a yellow text box, take it and place it in the project's application.properties file, and depending on which one, both will need it (more in “important classes and files” section).

1. Important Classes and files

WorkOrdersApplication.java - This is the main file, when you wish to run the program, this is the file you should look for

WorkOrdersController.java - This is the controller part of Maven, denotes the links in the project and can take data from the model and give it to the HTML to view, input, sotre, or anything else that will need to be done or acted on by the data.

Users.java - This is what sets the idea for a general user, if anything needs changed within the user (such has giving them a new property) then it will go here

Role.java - Creates a role id that is given and attached to users, also generates a users\_roles table that will show the relationship between the two of them

Incidents.java - holds information for the incidents that a user/admin/technician can create or send in, if anything needs to be added or removed, it would be here

WebSecurityConfig.java - is the file that makes sure that people accessing different URLs and aspects of the project have the proper authorization and ability to access it. Will work with the data.sql file to determine which roles a user can have and what pages they are allowed to access.

UserServices.java - This class will run when a user needs to be registered so they get added to the database, as well as attaching a default role and sending emails to users when they need an email.

IncidentServices.java - will save incidents so that the data can then be taken and shown to the user who has reported them.

Data.SQL - This will seed the database with some pre-determined information, data can be added and removed to suit the needs of people using the program, and works with the WebSecurityConfig file to determine which roles exist in which order (note that all users are given the default role of “User” located \*placeholder\*)

Application.properties - Holds critical information for running the program and there are two variations located within the project. Although the information that is shared between them is the SMTP information, which must also be configured with a viable Gmail and special application password.

* H2Application.properties.txt - replacing the information in this with what is in the normal file, will let it run with H2 compatibility, a memory-based database service that has no persistence, and will drop all information between each run.
* MySQLApplication.Properties.txt - using the information located in this file will give the project mySQL support and must be installed on the server to be used properly. Also note that there is a username and password which must be set to the username and password that is used to access mySQL on that server the project is being run

1. Interactions Between Classes
2. UML diagrams

This section will detail different UML diagrams for reference to this project, such as use and sequence diagrams

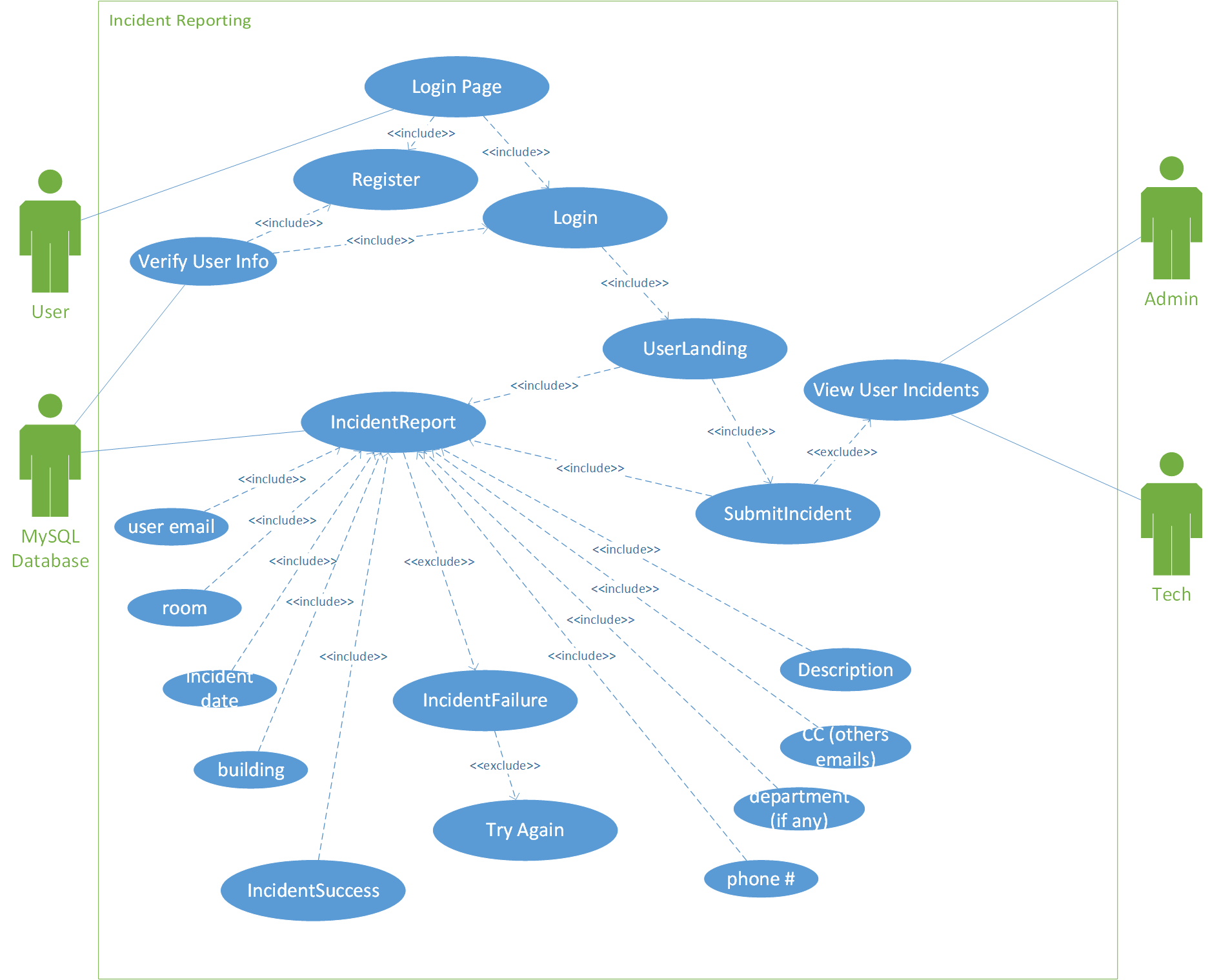


Figure 1. Use Case of Incident Reports

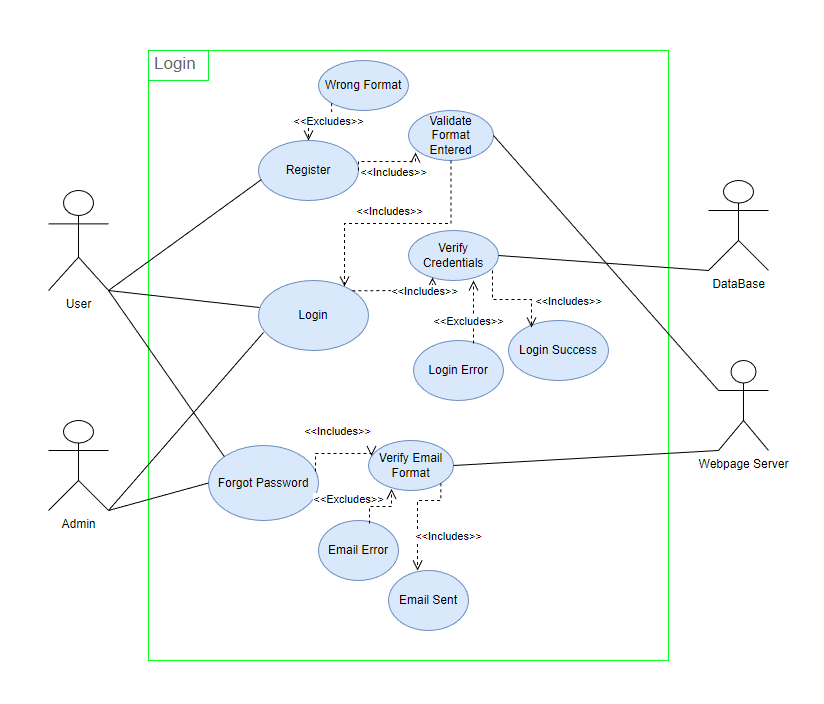
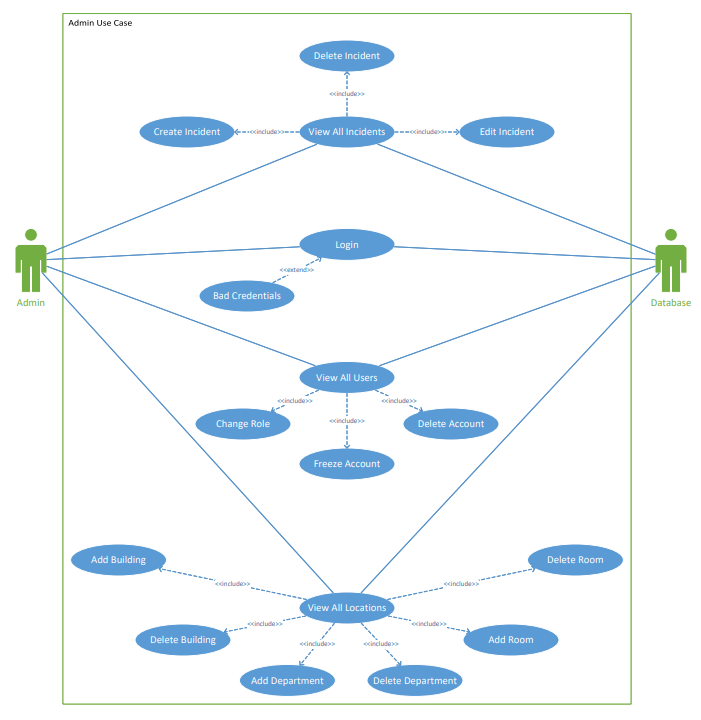


Figure 2. Use Case of Login

Figure 3. Use Case of Admins

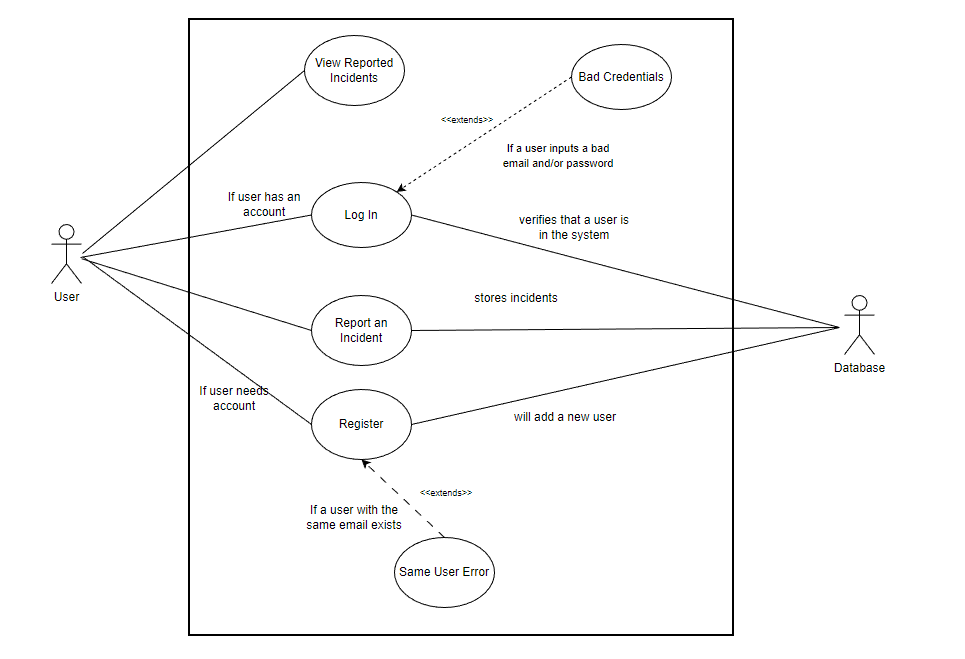


Figure 4. Use Case of a User Registering and creating an incident